A Brief Introduction to Bayesian Methods for Parameter Estimation, Model Tuning, and Uncertainty Quantification

Marcus van Lier-Walqui

CCSR Columbia University and NASA GISS

March 9, 2018

Motivation

You have some weather/climate model that you want to improve and some observations that might help you in some way. You might want to...

Motivation

You have some weather/climate model that you want to improve and some observations that might help you in some way. You might want to...

Parameter estimation

Find what model parameter values are "best", by some quantitative metric (which is likely not perfect in some way or another)

Motivation

You have some weather/climate model that you want to improve and some observations that might help you in some way. You might want to...

Parameter estimation

Find what model parameter values are "best", by some quantitative metric (which is likely not perfect in some way or another)

Sensitivity analysis

Figure out which model parameters are the "most important", i.e. have the greatest effect on some quantities of interest

Motivation

You have some weather/climate model that you want to improve and some observations that might help you in some way. You might want to. . .

Parameter estimation

Find what model parameter values are “best”, by some quantitative metric (which is likely not perfect in some way or another)

Sensitivity analysis

Figure out which model parameters are the “most important”, i.e. have the greatest effect on some quantities of interest

Uncertainty quantification

Get some measure for how *bad* your model is, or how little you know about its parameter values

Parameter Estimation

Overarching question: what is the most probable set of parameter values, given the information (theoretical, empirical, expert guess, etc.) available? Combining information this way can be. . .

Parameter Estimation

Overarching question: what is the most probable set of parameter values, given the information (theoretical, empirical, expert guess, etc.) available? Combining information this way can be...

Express probabilistically

What is the probability of some parameter value $\mathbf{x}$ given some new information (data) $\mathbf{y}$, or... $P(\mathbf{x}|\mathbf{y})$

Parameter Estimation

Overarching question: what is the most probable set of parameter values, given the information (theoretical, empirical, expert guess, etc.) available? Combining information this way can be...

Express probabilistically

What is the probability of some parameter value $\mathbf{x}$ given some new information (data) $\mathbf{y}$, or... $P(\mathbf{x}|\mathbf{y})$

Bayes' theorem

$$P(\mathbf{x}|\mathbf{y}, M) = \frac{P(\mathbf{x}|M) \cdot P(\mathbf{y}|\mathbf{x}, M)}{P(\mathbf{y}|M)} \tag{1}$$

Parameter Estimation

Overarching question: what is the most probable set of parameter values, given the information (theoretical, empirical, expert guess, etc.) available? Combining information this way can be...

Express probabilistically

What is the probability of some parameter value $\mathbf{x}$ given some new information (data) $\mathbf{y}$, or... $P(\mathbf{x}|\mathbf{y})$

Bayes' theorem

$$P(\mathbf{x}|\mathbf{y}, M) = \frac{P(\mathbf{x}|M) \cdot P(\mathbf{y}|\mathbf{x}, M)}{P(\mathbf{y}|M)} \tag{1}$$

- $P(\mathbf{x}|M)$ – prior PDF of control parameters
- $P(\mathbf{y}|\mathbf{x}, M)$ – likelihood of observations given parameter values
- All probabilities are conditional on the choice of model M!

Solving Bayes: Complete Enumeration 1

One can simply span the full parameter space and map out the probability of all possibilities.
e.g. Lets say we have 5 data points and want to fit a Gaussian, what is the probability of a particular choice of μ,σ?

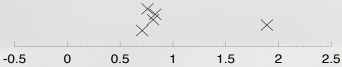


Figure from MacKay (2005)

Complete Enumeration 2

Set bounds and discretize space in μ, σ dimensions

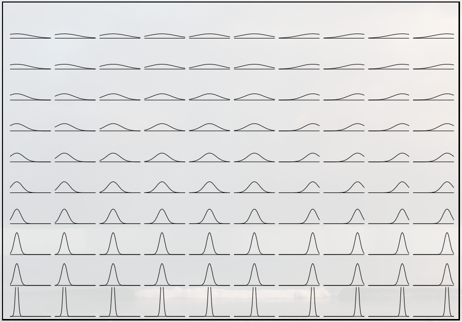


Figure from MacKay (2005)

Complete Enumeration 3

Calculate probability of parameters given data (shown via thickness of lines, with very small probability not shown).

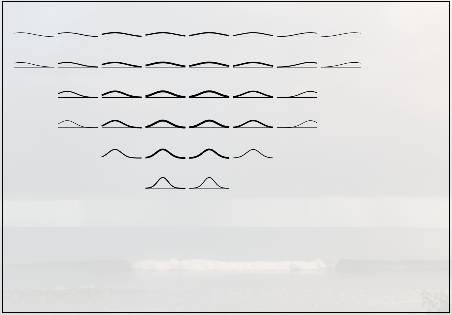


Figure from MacKay (2005)

Complete Enumeration 4

Advantages:

- Works for any distribution (no Gaussian assumption)
- Easy to code
- Efficiency depends only on resolution and number of parameter dimensions

Disadvantages

- No clear way to efficiently/adequately span parameter space
- Curse of dimensionality (cost increases with dimension as $(N_x)^n$)

Kalman Filter/Smoother

Assuming all probabilities are Gaussian and your model is linear (i.e. can be expressed as a matrix), Bayes' theorem is trivial to solve:

Kalman Filter/Smoother

Assuming all probabilities are Gaussian and your model is linear (i.e. can be expressed as a matrix), Bayes' theorem is trivial to solve:

$$\overline{\mathbf{x}^a} = \overline{\mathbf{x}^f} + \mathbf{P}^f\mathbf{H}^T(\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R})^{-1}\left[\mathbf{y} - \overline{\mathbf{H}\mathbf{x}^f}\right],$$

Kalman Filter/Smoother

Assuming all probabilities are Gaussian and your model is linear (i.e. can be expressed as a matrix), Bayes' theorem is trivial to solve:

$$\overline{\mathbf{x}^a} = \overline{\mathbf{x}^f} + \mathbf{P}^f\mathbf{H}^T(\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R})^{-1}\left[\mathbf{y} - \overline{\mathbf{H}\mathbf{x}^f}\right],$$

Where $\mathbf{x}^a$ is the posterior mean, $\mathbf{H}$ is the model matrix, $\mathbf{P}^{\mathbf{f}}$ is the forecast covariance and $\mathbf{R}$ is the observational error covariance.

Kalman Filter/Smoother

Assuming all probabilities are Gaussian and your model is linear (i.e. can be expressed as a matrix), Bayes' theorem is trivial to solve:

$$\overline{\mathbf{x}^a} = \overline{\mathbf{x}^f} + \mathbf{P}^f\mathbf{H}^T(\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R})^{-1}\left[\mathbf{y} - \overline{\mathbf{H}\mathbf{x}^f}\right],$$

Where $\mathbf{x}^a$ is the posterior mean, $\mathbf{H}$ is the model matrix, $\mathbf{P}^\mathbf{f}$ is the forecast covariance and $\mathbf{R}$ is the observational error covariance.
For nonlinear models, ensemble approximations of terms in the Kalman filter are used, yielding the *ensemble* Kalman filter (EnKF)

Kalman Filter/Smoother

Assuming all probabilities are Gaussian and your model is linear (i.e. can be expressed as a matrix), Bayes' theorem is trivial to solve:

$$\overline{\mathbf{x}^a} = \overline{\mathbf{x}^f} + \mathbf{P}^f\mathbf{H}^T(\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R})^{-1}\left[\mathbf{y} - \overline{\mathbf{H}\mathbf{x}^f}\right],$$

Where $\mathbf{x}^a$ is the posterior mean, $\mathbf{H}$ is the model matrix, $\mathbf{P}^\mathbf{f}$ is the forecast covariance and $\mathbf{R}$ is the observational error covariance. For nonlinear models, ensemble approximations of terms in the Kalman filter are used, yielding the *ensemble* Kalman filter (EnKF) For strongly nonlinear problems (most parameter estimation problems), these approximations are *really bad*.

Markov chain Monte-Carlo (MCMC)

We *intelligently* sample the parameter space:

- Use a modified random walk (a Markov chain) to sample the parameter space

Markov chain Monte-Carlo (MCMC)

We *intelligently* sample the parameter space:

- Use a modified random walk (a Markov chain) to sample the parameter space
- Random walk can be Gaussian or uniform (or anything else)

Markov chain Monte-Carlo (MCMC)

We *intelligently* sample the parameter space:

- Use a modified random walk (a Markov chain) to sample the parameter space
- Random walk can be Gaussian or uniform (or anything else)
- Each new sample depends *only* on the previous sample (Markovian property).

Markov chain Monte-Carlo (MCMC)

We *intelligently* sample the parameter space:

- Use a modified random walk (a Markov chain) to sample the parameter space
- Random walk can be Gaussian or uniform (or anything else)
- Each new sample depends *only* on the previous sample (Markovian property).
- Each new sample is accepted or rejected depending on probabilities of prior/proposal:

Markov chain Monte-Carlo (MCMC)

We *intelligently* sample the parameter space:

- Use a modified random walk (a Markov chain) to sample the parameter space
- Random walk can be Gaussian or uniform (or anything else)
- Each new sample depends *only* on the previous sample (Markovian property).
- Each new sample is accepted or rejected depending on probabilities of prior/proposal:

$$P(\mathbf{x}_{prop}|\mathbf{x}_{prior}) = min[1, P(\mathbf{x}_{prop})/P(\mathbf{x}_{prior})]$$

Markov chain Monte-Carlo (MCMC)

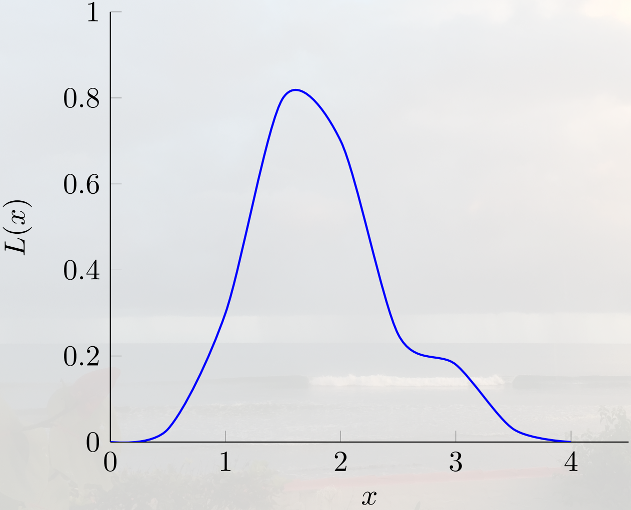
We *intelligently* sample the parameter space:

- Use a modified random walk (a Markov chain) to sample the parameter space
- Random walk can be Gaussian or uniform (or anything else)
- Each new sample depends *only* on the previous sample (Markovian property).
- Each new sample is accepted or rejected depending on probabilities of prior/proposal:

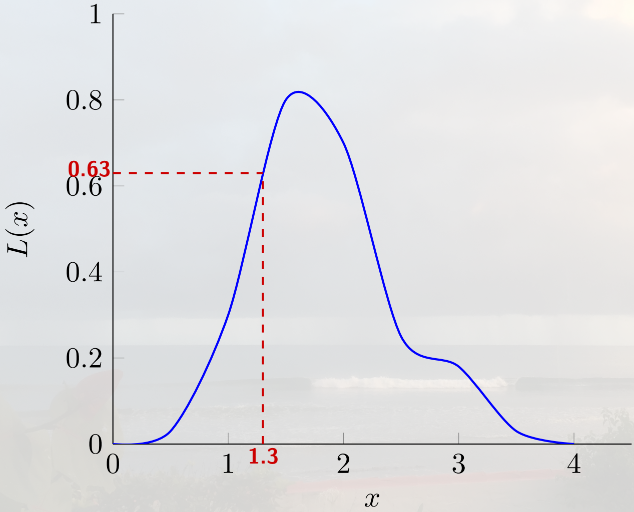
$$P(\mathbf{x}_{prop}|\mathbf{x}_{prior}) = min[1, P(\mathbf{x}_{prop})/P(\mathbf{x}_{prior})]$$

The density of samples matches $P(\mathbf{x}|\mathbf{y}, M)$

Markov chain example - the Metropolis sampler



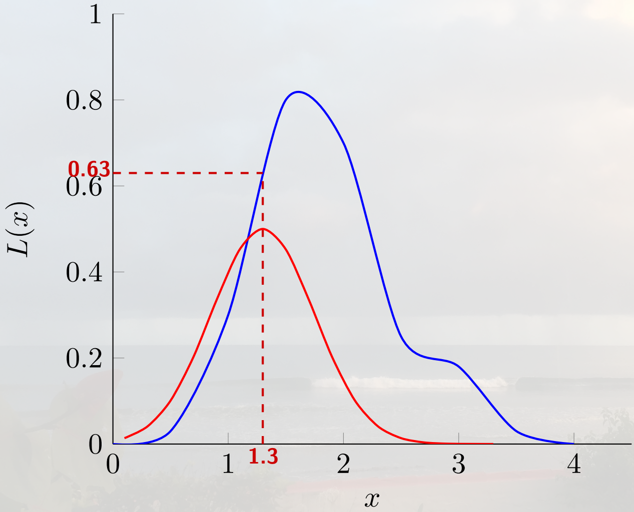
Markov chain example - the Metropolis sampler



Markov Chain

i	x(i)
1	1.3

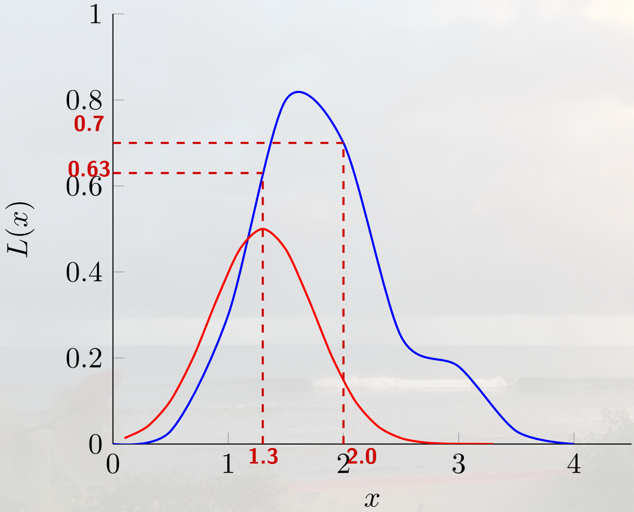
Markov chain example - the Metropolis sampler



Markov Chain

i	x(i)
1	1.3

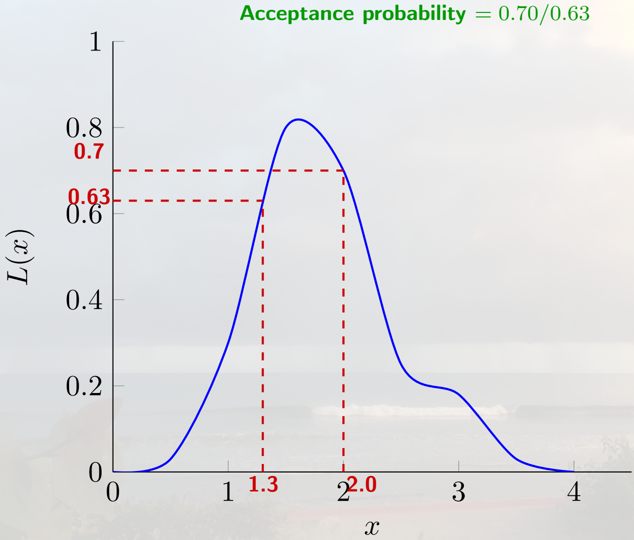
Markov chain example - the Metropolis sampler



Markov Chain

i	x(i)
1	1.3

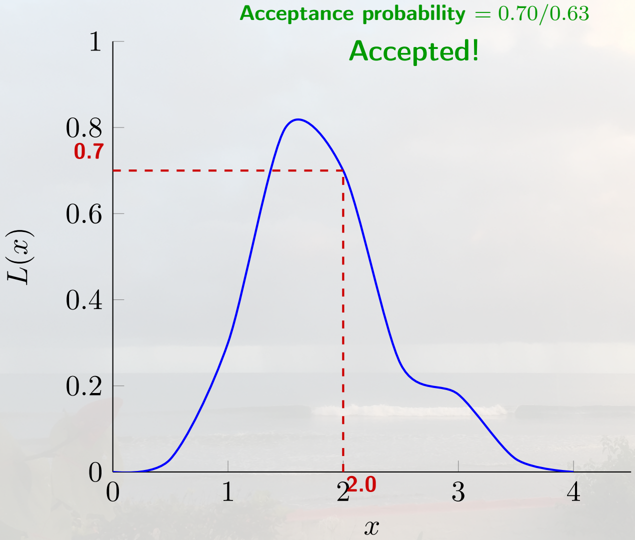
Markov chain example - the Metropolis sampler



Markov Chain

i	x(i)
1	1.3

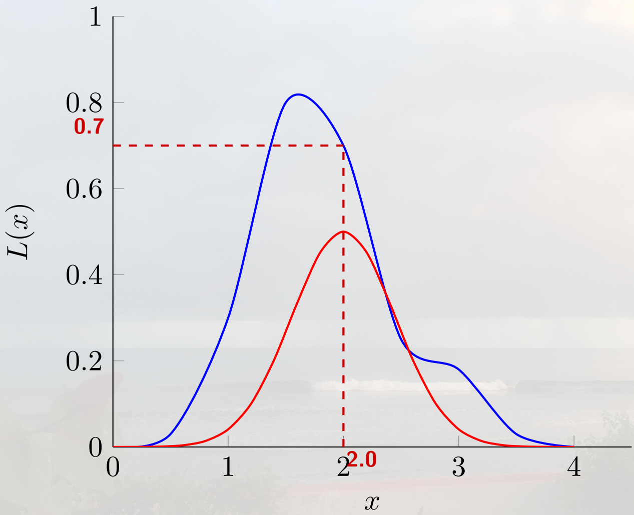
Markov chain example - the Metropolis sampler



Markov Chain

i	x(i)
1	1.3
2	2.0

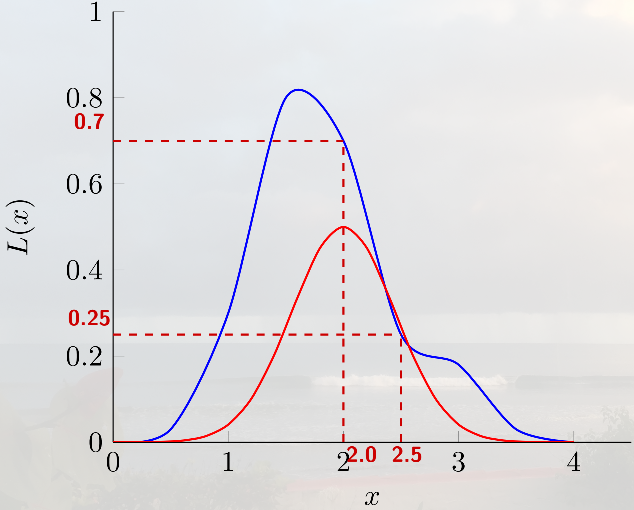
Markov chain example - the Metropolis sampler



Markov Chain

i	x(i)
1	1.3
2	2.0

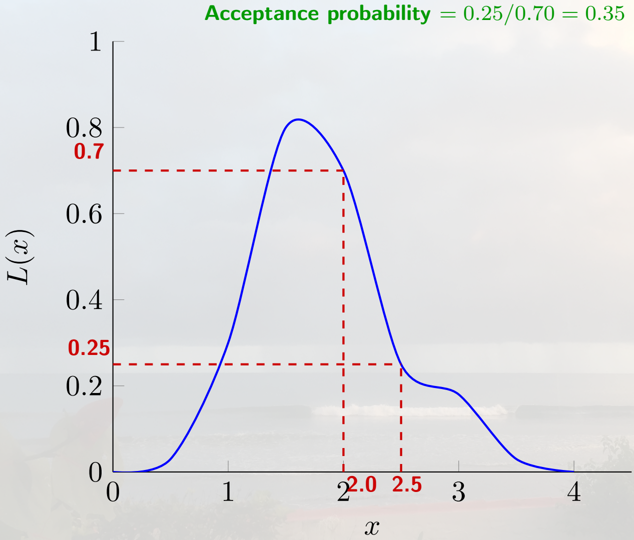
Markov chain example - the Metropolis sampler



Markov Chain

i	x(i)
1	1.3
2	2.0

Markov chain example - the Metropolis sampler

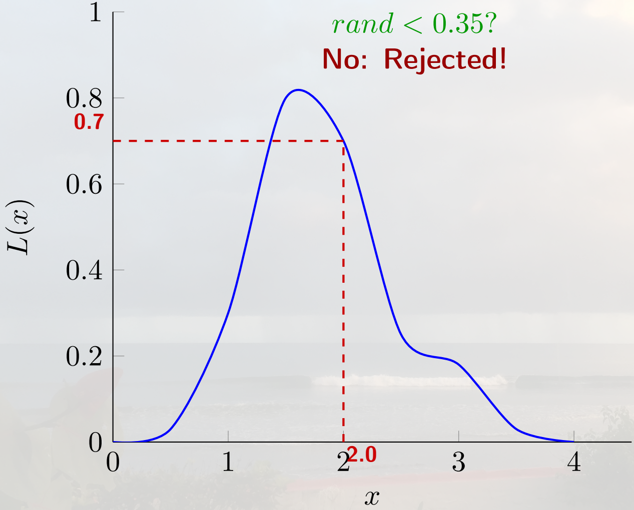


Markov Chain

i	x(i)
1	1.3
2	2.0

Markov chain example - the Metropolis sampler

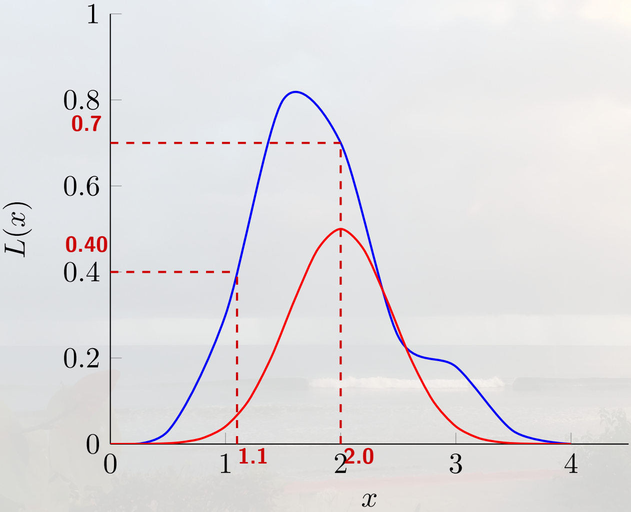
Acceptance probability $= 0.25/0.70 = 0.35$



Markov Chain

i	x(i)
1	1.3
2	2.0
3	2.0

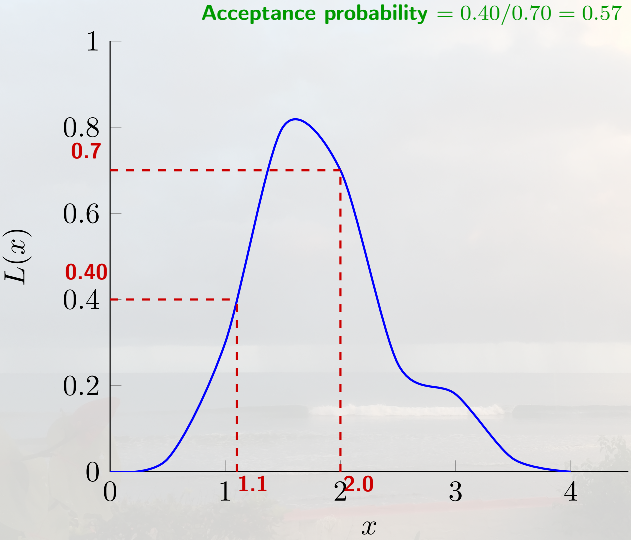
Markov chain example - the Metropolis sampler



Markov Chain

i	x(i)
1	1.3
2	2.0
3	2.0

Markov chain example - the Metropolis sampler



Markov Chain

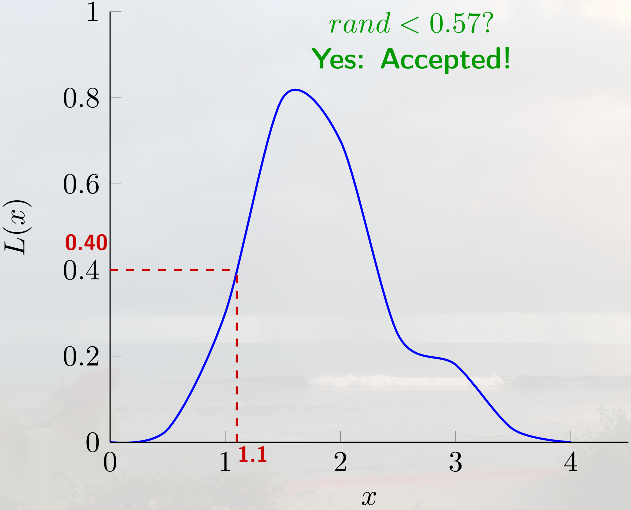
i	x(i)
1	1.3
2	2.0
3	2.0

Markov chain example - the Metropolis sampler

Acceptance probability $= 0.40/0.70 = 0.57$

$rand < 0.57?$

Yes: Accepted!



Markov Chain

i	x(i)
1	1.3
2	2.0
3	2.0
4	1.1

MCMC in action

Practical issues with MCMC

- No efficient way to parallelize

Practical issues with MCMC

- No efficient way to parallelize
- Assessing convergence can be tricky

Practical issues with MCMC

- No efficient way to parallelize
- Assessing convergence can be tricky
- Requires zillions of samples (model integrations!)

Practical issues with MCMC

- No efficient way to parallelize
- Assessing convergence can be tricky
- Requires zillions of samples (model integrations!)
- Relies on accurate prior and observational uncertainty

Practical issues with MCMC

- No efficient way to parallelize
- Assessing convergence can be tricky
- Requires zillions of samples (model integrations!)
- Relies on accurate prior and observational uncertainty
- Assumes that the parameters of interest are the *main* source of uncertainty

Practical issues with MCMC

- No efficient way to parallelize
- Assessing convergence can be tricky
- Requires zillions of samples (model integrations!)
- Relies on accurate prior and observational uncertainty
- Assumes that the parameters of interest are the *main* source of uncertainty

The Bottom Line:

MCMC methods are great for tricky (strongly nonlinear, multimodal, ill-posed) parameter estimation problems where model integration is relatively cheap. Even then, they require care and expert guidance (model/observation).

Simulated Annealing 1

For optimization problems, we can modify a MCMC sampler to more efficiently find high-probability regions of the parameter space.

Simulated Annealing 1

For optimization problems, we can modify a MCMC sampler to more efficiently find high-probability regions of the parameter space.

- Perform MCMC walk, similarly to Metropolis sampling

Simulated Annealing 1

For optimization problems, we can modify a MCMC sampler to more efficiently find high-probability regions of the parameter space.

- Perform MCMC walk, similarly to Metropolis sampling
- Scale the transition probability by a "temperature" which decreases with sample size.

Simulated Annealing 1

For optimization problems, we can modify a MCMC sampler to more efficiently find high-probability regions of the parameter space.

- Perform MCMC walk, similarly to Metropolis sampling
- Scale the transition probability by a "temperature" which decreases with sample size.
- This allows for bold transitions when the sampler is "hot" and more conservative transitions when the sampler is "cold"

Simulated Annealing 1

For optimization problems, we can modify a MCMC sampler to more efficiently find high-probability regions of the parameter space.

- Perform MCMC walk, similarly to Metropolis sampling
- Scale the transition probability by a "temperature" which decreases with sample size.
- This allows for bold transitions when the sampler is "hot" and more conservative transitions when the sampler is "cold"

For example. . .

Simulated Annealing 1

For optimization problems, we can modify a MCMC sampler to more efficiently find high-probability regions of the parameter space.

- Perform MCMC walk, similarly to Metropolis sampling
- Scale the transition probability by a "temperature" which decreases with sample size.
- This allows for bold transitions when the sampler is "hot" and more conservative transitions when the sampler is "cold"

For example. . .

$$P_{SA} = P^{\frac{1}{T}}$$

Simulated Annealing 1

For optimization problems, we can modify a MCMC sampler to more efficiently find high-probability regions of the parameter space.

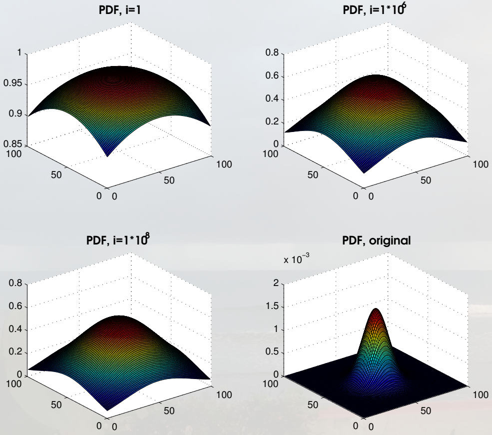
- Perform MCMC walk, similarly to Metropolis sampling
- Scale the transition probability by a "temperature" which decreases with sample size.
- This allows for bold transitions when the sampler is "hot" and more conservative transitions when the sampler is "cold"

For example...

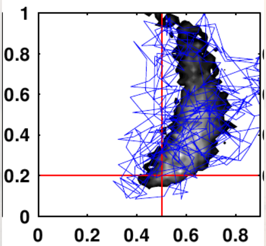
$$P_{SA} = P^{\frac{1}{T}}$$

$$T_i = \frac{200}{\log(i+1)}$$

Simulated Annealing 2



Simulated annealing used to pre-sample before running Metropolis MCMC:



Gibbs Sampling

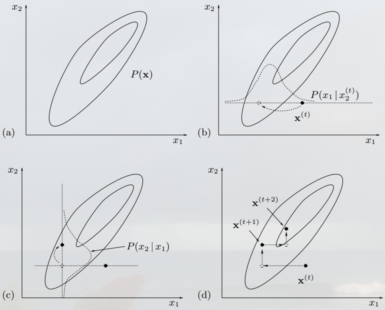


Figure from MacKay [2005]

- What if you can sample from the conditional distribution?
- Take turns sampling from conditionals of each dimension
- Acceptance ratio = 1 (always!)
- Freely available software (BUGS) - Bayesian inference Using Gibbs Sampling

Other Monte Carlo topics

- Hamiltonian (hybrid) MCMC and No U-Turn Sampler
- Affine-invariant MCMC (The MCMC Hammer)
- Importance sampling
- Slice sampler
- Perfect sampler
- Nested (& multimodal nested sampling)
- MC methods for model comparison (estimation of ‘evidence’)
- Particle filter
- Ensemble Kalman Filter

Surrogate techniques

What if the model is still too expensive, and you can only afford to run it 1000, 500, or 100 times?

Surrogate techniques

What if the model is still too expensive, and you can only afford to run it 1000, 500, or 100 times?

- Emulate the response of some model to perturbation of parameters by nonlinear regression

Surrogate techniques

What if the model is still too expensive, and you can only afford to run it 1000, 500, or 100 times?

- Emulate the response of some model to perturbation of parameters by nonlinear regression
- Perform parameter estimation or sensitivity analysis or UQ on the (cheap!) surrogate model rather than the full model

Surrogate techniques

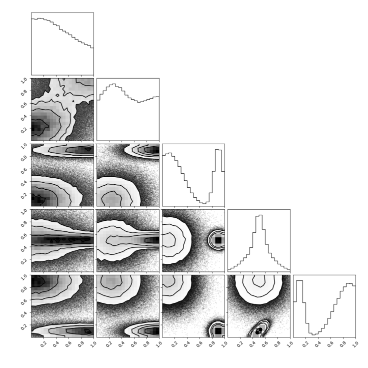
What if the model is still too expensive, and you can only afford to run it 1000, 500, or 100 times?

- Emulate the response of some model to perturbation of parameters by nonlinear regression
- Perform parameter estimation or sensitivity analysis or UQ on the (cheap!) surrogate model rather than the full model
- Choices: Gaussian Process Models, Polynomial Chaos Expansion, etc.

Surrogate techniques

What if the model is still too expensive, and you can only afford to run it 1000, 500, or 100 times?

- Emulate the response of some model to perturbation of parameters by nonlinear regression
- Perform parameter estimation or sensitivity analysis or UQ on the (cheap!) surrogate model rather than the full model
- Choices: Gaussian Process Models, Polynomial Chaos Expansion, etc.

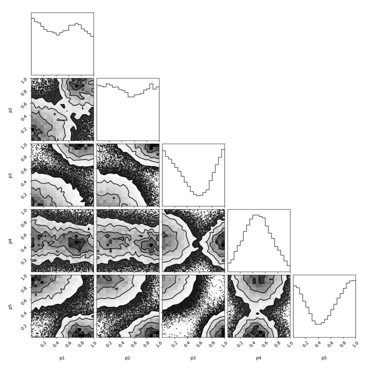


about 500,000 samples

Surrogate techniques

What if the model is still too expensive, and you can only afford to run it 1000, 500, or 100 times?

- Emulate the response of some model to perturbation of parameters by nonlinear regression
- Perform parameter estimation or sensitivity analysis or UQ on the (cheap!) surrogate model rather than the full model
- Choices: Gaussian Process Models, Polynomial Chaos Expansion, etc.

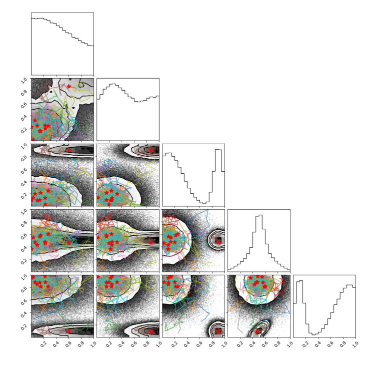


only 500 samples

Surrogate techniques

What if the model is still too expensive, and you can only afford to run it 1000, 500, or 100 times?

- Emulate the response of some model to perturbation of parameters by nonlinear regression
- Perform parameter estimation or sensitivity analysis or UQ on the (cheap!) surrogate model rather than the full model
- Choices: Gaussian Process Models, Polynomial Chaos Expansion, etc.



Sim. Annealing: 500×10 samples

Estimating Ice Microphysics Parameters

Oue et al JAMC 2016

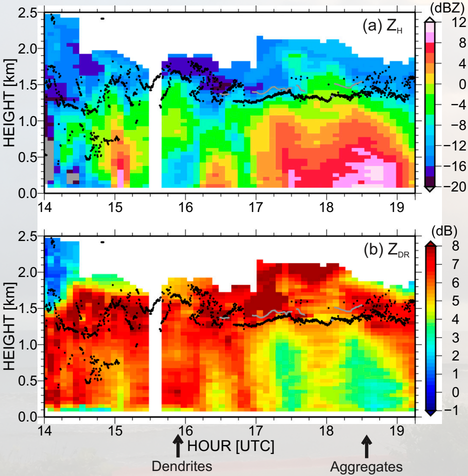


FIG. 7. Time-vs-height cross sections of the X-SAPR (a) Z_H and (b) Z_{DR} on 2 May 2013. Each profile represents the mean values of all points with elevation angles of 14°–15° (165°–166°) in 50-m height increments from three HRHI scans (azimuth angles of 7°, 52°, and 97°) every approximately 5 min. The horizontal gray lines and black dots respectively represent liquid-cloud top estimated from the KAZR Doppler spectrum width and cloud base observed by a ceilometer.

Estimating Ice Microphysics Parameters

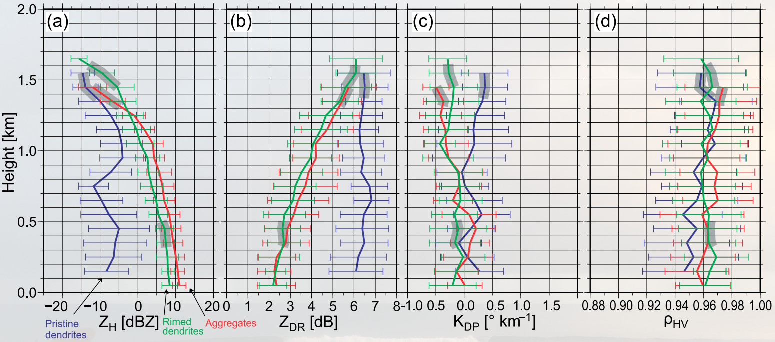
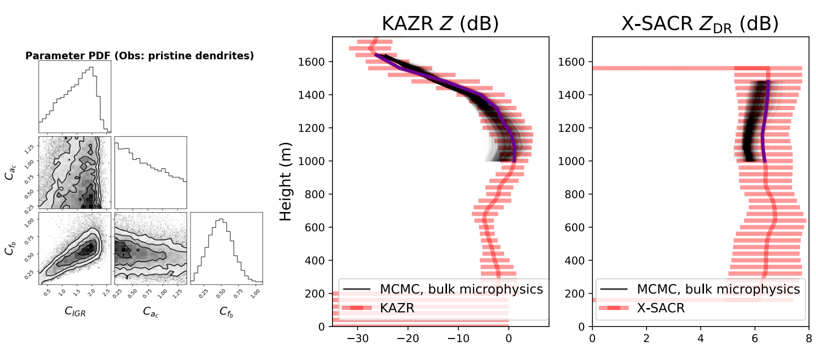


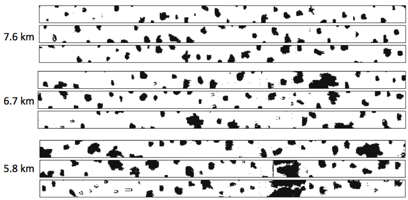
FIG. 17. Vertical profiles of averaged (a) Z_H, (b) Z_{DR}, (c) K_{DP}, and (d) ρ_{HV} from the X-SAPR HRHIs, during which the pristine dendrites (blue line), aggregates (red line), and rimed dendrites (green line) were observed at the ground. The averaging areas are presented in Figs. 6, 9, and 13. Averages were calculated in 100-m altitude increments from all values with elevation angles <20° or >160°. The total number of samples in each profile exceeds 1900. Error bars represent standard deviations. Gray shading represents layers between ceilometer-measured cloud base and topmost liquid-cloud top.

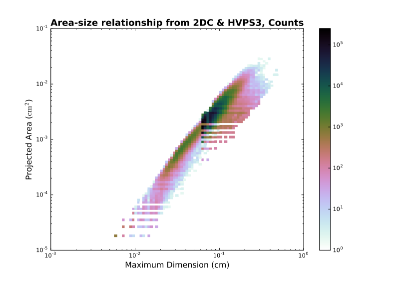
Estimating Ice Microphysics Parameters



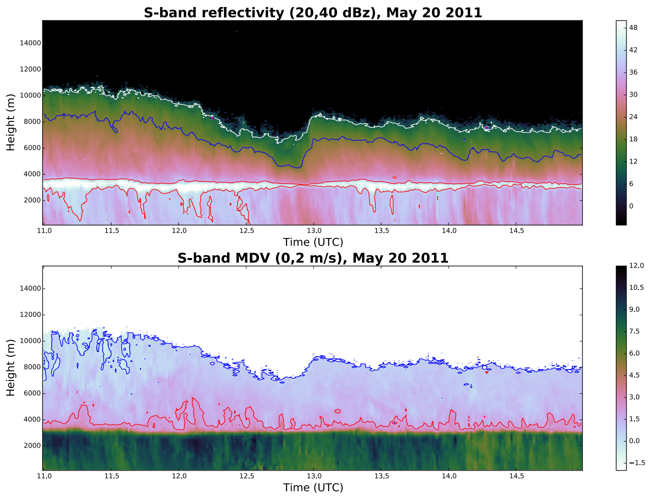
More Ice Microphysics: Aggregation

In situ (2DC & HVPS)



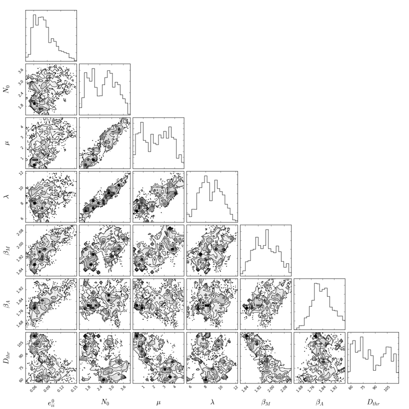


More Ice Microphysics: Aggregation

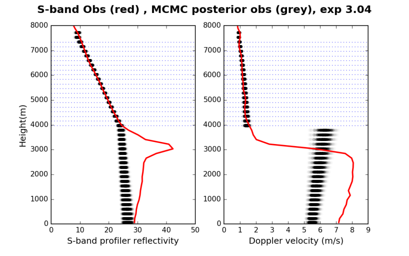


More Ice Microphysics: Aggregation

Sticking efficiency *and* ice property/PSD

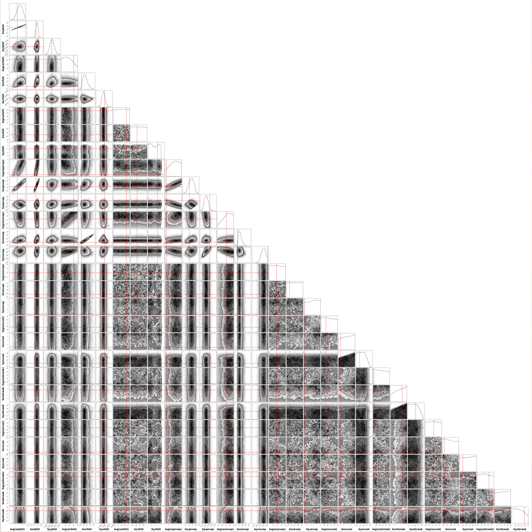


Fwd-simulated Z, MDV profiles

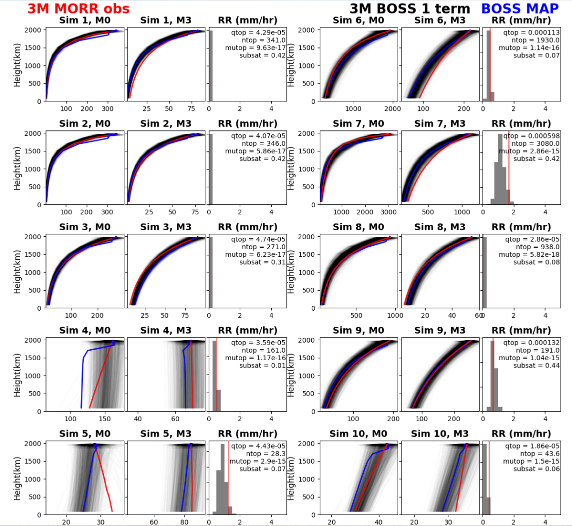


A Probabilistic Microphsyics Scheme

Bayesian
Observationally-
constrained
Statistical-physical
Scheme (BOSS)



A Probabilistic Microphysics Scheme

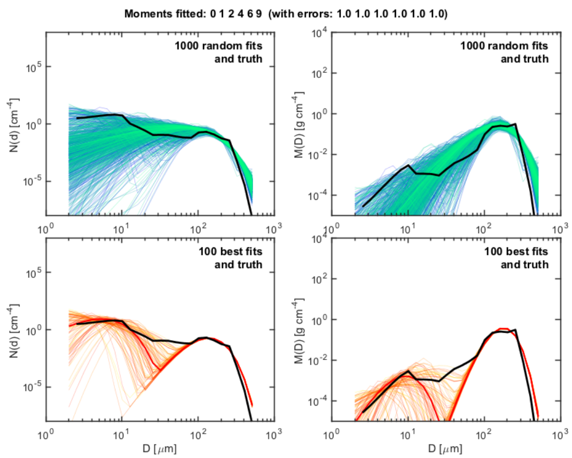


3-moment performs better than 2-moment BOSS (higher likelihood)

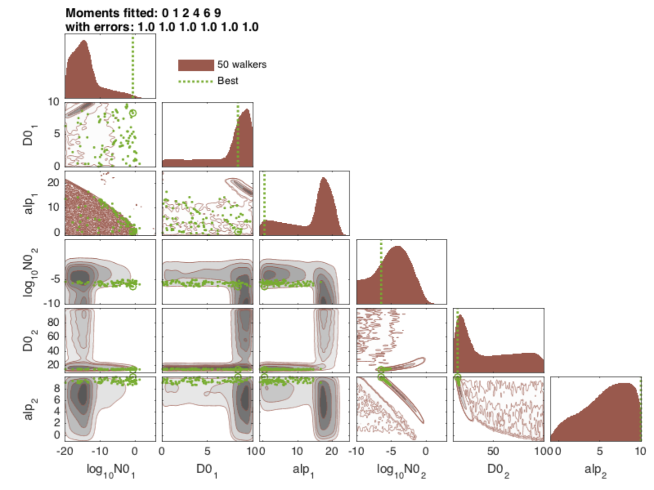
Predicted uncertainty is a good match to error in almost all cases.

Some issue with (numerical) oscillation at top-of-rain shaft

Cloud Property Retrieval using Radar



Cloud Property Retrieval using Radar



Sensitivity analysis

The end

Thanks for listening!

What is the likelihood?

$$P(\mathbf{x}|\mathbf{y}) = \frac{P(\mathbf{x}) \cdot P(\mathbf{y}|\mathbf{x})}{P(\mathbf{y})} \tag{2}$$

What is the likelihood?

$$P(\mathbf{x}|\mathbf{y}) = \frac{P(\mathbf{x}) \cdot P(\mathbf{y}|\mathbf{x})}{P(\mathbf{y})} \tag{2}$$

Assuming Gaussian error in our observations, the likelihood is:

What is the likelihood?

$$P(\mathbf{x}|\mathbf{y}) = \frac{P(\mathbf{x}) \cdot P(\mathbf{y}|\mathbf{x})}{P(\mathbf{y})} \tag{2}$$

Assuming Gaussian error in our observations, the likelihood is:

$$P(\mathbf{y}|\mathbf{x}) = e^{-\Phi_{\mathbf{xy}}}, \tag{3}$$

What is the likelihood?

$$P(\mathbf{x}|\mathbf{y}) = \frac{P(\mathbf{x}) \cdot P(\mathbf{y}|\mathbf{x})}{P(\mathbf{y})} \tag{2}$$

Assuming Gaussian error in our observations, the likelihood is:

$$P(\mathbf{y}|\mathbf{x}) = e^{-\Phi_{\mathbf{xy}}}, \tag{3}$$

$$\Phi_{\mathbf{xy}} = \frac{1}{2}(f(\mathbf{x}) - \mathbf{y})^{\mathbf{T}}\mathbf{C}^{-1}(f(\mathbf{x}) - \mathbf{y}) \tag{4}$$

What is the likelihood?

$$P(\mathbf{x}|\mathbf{y}) = \frac{P(\mathbf{x}) \cdot P(\mathbf{y}|\mathbf{x})}{P(\mathbf{y})} \tag{2}$$

Assuming Gaussian error in our observations, the likelihood is:

$$P(\mathbf{y}|\mathbf{x}) = e^{-\Phi_{\mathbf{xy}}}, \tag{3}$$

$$\Phi_{\mathbf{xy}} = \frac{1}{2}(f(\mathbf{x}) - \mathbf{y})^{\mathbf{T}}\mathbf{C}^{-1}(f(\mathbf{x}) - \mathbf{y}) \tag{4}$$

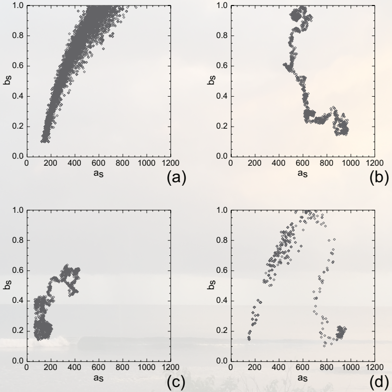
$f(\mathbf{x})$ is result of propagating the control parameters $\mathbf{x}$ through the forward model f.
$\mathbf{y}$ is the (true) observational vector.
$\mathbf{C}$ is the observation error covariance matrix.

Practical issues with MCMC: Proposal issues 1

Poorly tuned proposal distribution can cause problems. Also, bad choice of start position can be problematic.

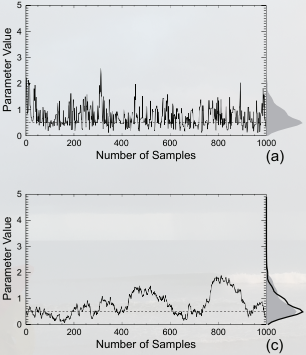
- A: Good proposal variance
- B: Proposal variance small, started far from large PDF values
- C: same as B, started within region of large PDF values
- D: Same as B, adaptive proposal variance

Figures from Posselt [2012]



Practical issues with MCMC: Proposal issues 2

Time series of chain can show problematic autocorrelation due to poorly chosen proposal and/or non-covergent sample.



Figures from Posselt [2012]

Practical issues with MCMC: Proposal issues 3

How does one construct a good proposal?

How does one avoid bad start position?

- Prior knowledge
- Run many chains with random start positions
- Run simulated annealing "pre-sampler"

Practical issues with MCMC: Proposal issues 3

How does one construct a good proposal?

- Prior knowledge

How does one avoid bad start position?

- Prior knowledge
- Run many chains with random start positions
- Run simulated annealing "pre-sampler"

Practical issues with MCMC: Proposal issues 3

How does one construct a good proposal?

- Prior knowledge
- "Burn-in" phase where proposal is actively tuned

How does one avoid bad start position?

- Prior knowledge
- Run many chains with random start positions
- Run simulated annealing "pre-sampler"

Practical issues with MCMC: Proposal issues 3

How does one construct a good proposal?

- Prior knowledge
- "Burn-in" phase where proposal is actively tuned
- Adaptive Metropolis (proposal variance constantly tuned)

How does one avoid bad start position?

- Prior knowledge
- Run many chains with random start positions
- Run simulated annealing "pre-sampler"

Practical issues with MCMC: Proposal issues 3

How does one construct a good proposal?

- Prior knowledge
- "Burn-in" phase where proposal is actively tuned
- Adaptive Metropolis (proposal variance constantly tuned)
- Delayed Rejection (2nd proposal after 1st)

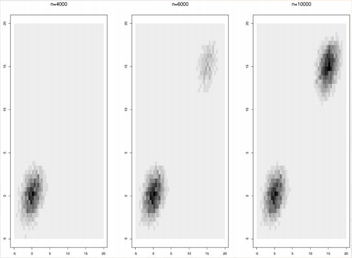
How does one avoid bad start position?

- Prior knowledge
- Run many chains with random start positions
- Run simulated annealing "pre-sampler"

Practical issues with MCMC: Asessing convergence 1

When do we stop our chain? How do we tell if we've converged to the target PDF?

- If the target distribution is known, compare
- Assess convergence of running statistical moments
- Kolmogorov-Smirnov test on chain sub-samples
- R-statistic – Gelman et al. [1996]
- *Caveat:* beware of 'pseudo-convergence'!



Practical issues with MCMC: Asessing convergence 2

R-Statistic – Gelman et al. [1996]

General idea:

- Run many chains
- Compute variance within each chain (W)
- Compute mean of each chain
- Compare mean of within-chain variances with variance of all chain means (B)

$$\hat{var}^{+}(\mathbf{x}|\mathbf{y}) = \frac{n-1}{n}W + \frac{1}{n}B \tag{5}$$

$$\hat{R} = \sqrt{\frac{\hat{var}^{+}(\mathbf{x}|\mathbf{y})}{W}} \tag{6}$$

Summary

- Monte Carlo methods can solve tough inference problems using random numbers

Summary

- Monte Carlo methods can solve tough inference problems using random numbers
- Much cheaper than complete enumeration, especially as dimensions increase

Summary

- Monte Carlo methods can solve tough inference problems using random numbers
- Much cheaper than complete enumeration, especially as dimensions increase
- Robust, make no assumptions of model linearity or PDF Gaussianity

Summary

- Monte Carlo methods can solve tough inference problems using random numbers
- Much cheaper than complete enumeration, especially as dimensions increase
- Robust, make no assumptions of model linearity or PDF Gaussianity
- Require many model integrations

Summary

- Monte Carlo methods can solve tough inference problems using random numbers
- Much cheaper than complete enumeration, especially as dimensions increase
- Robust, make no assumptions of model linearity or PDF Gaussianity
- Require many model integrations
- Often do not parallelize well

Summary

- Monte Carlo methods can solve tough inference problems using random numbers
- Much cheaper than complete enumeration, especially as dimensions increase
- Robust, make no assumptions of model linearity or PDF Gaussianity
- Require many model integrations
- Often do not parallelize well
- For more info see:
    - Tarantola [2005]
    - MacKay [2005]
    - Robert and Casella

References I

A. Gelman, G. O. Roberts, and W. R. Gilks. Efficient metropolis jumping rules. *Bayesian Statistics*, 5:599–607, 1996.

D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, Cambridge, UK, 7.2 edition, 2005.

D. J. Posselt. *Markov chain Monte Carlo Mehtods: Theory and Applications. Data Assimilation for Atmospheric, Oceanic and Hydrologic Applications*. Springer, In Press., 2 edition, 2012.

A. Tarantola. *Inverse Problem Theory*. SIAM, Philadelphia, 2005.